Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A method for locating neighbor objects of a query geometry in a database, the method comprising:

1) tiling a query geometry at a tiling level t;

2) conducting a search of a search area comprising interior and boundary tiles of the query geometry for neighbors among objects in the database, wherein the objects in the database are indexed using a linear quadtree index, if the search does not produce a result set that includes a target number of neighbors, then the following steps are carried out starting with a radius r of 1:

3) expanding the search area to a radius r tiles at a distance d from the boundary tiles and conducting a search for neighbors in the expanded search area, where $(r-1)*tw < d \leq r*tw$, wherein tw is the maximum or minimum of the x-width, y-width of a tile at level t and r is the query radius;

4) using each tile in the expanded search area it is determined if there are any geometries at distance $\leq r*tw$ that are to be included in the result set;

5) if a target number of neighbors is not found, steps 3-4 are repeated with a query radius of 2 tile widths followed by 3 tile widths;

6) if the target number still is not reached retiling the query geometry at a higher level of tiling; and

7) repeating steps 1-7 until either the target number is reached or all objects in the database are searched.

2. (original) The method according to claim 1, wherein geometries at distances >r*tw are included in the search with query radius r.

3. (original) The method according to claim 1, wherein only a portion of tile in the expanded query radius are materialized prior to searching the expanded query radius for neighbors, and wherein additional tiles are materialized until all tiles at a specific query radius have been generated, searched and included in a result.

4. (original) The method according to claim 3, wherein about 1000 to about 5000 tiles are materialized prior to searching.

5. (original) The method according to claim 3, further comprising removing duplicate neighbors from the result.

6. (original) The method according to claim 3, further comprising:

retaining only the target number of neighbors if more than the target number of neighbors is found and terminating the searching process.

7. (currently amended) A method for locating neighbor objects of a query object in a database, the method comprising:

defining a query tile set comprising a plurality of tiles that define an initial query radius that the query object is grouped within [[;]] , wherein the query object is a non-point object;

locating neighbor objects among objects in the database within the initial query radius [[;]] , wherein the objects in the database are indexed using a linear quadtree index, and wherein the objects in the database include a non-point object;

determining a number of neighbor objects within the initial query radius;

expanding the query radius beyond the initial query radius by defining an expanded query tile set beyond the query tile set if the number of neighbor objects is less than a target number;

locating neighbor objects within the expanded query radius;

determining if a number of neighbor objects within the expanded radius corresponds to a target number; and

continuing to expand the query radius and locate neighbor objects within the expanded radius until the number of neighbor objects equals or exceeds the target number.

8. (original) The method according to claim 7, further comprising:

calculating a distance of each neighbor object from the query object.

9. (original) The method according to claim 8, wherein the distance of each neighbor object from the center of the query object is calculated.

10. (currently amended) The method according to claim [[7]] 8, wherein the distance is based upon interboundary distance, boundary-to-boundary distance, or touch or intersect distance.

11. (currently amended) The method according to claim 7, further comprising:

redefining the initial query tile set as larger tiles to define a revised initial query radius if the target number of neighbor objects is not found in a minimum number of expansions of the query radius;

locating neighbor objects within the revised initial query radius;

determining a number of neighbor objects within the revised initial query radius;

defining a revised expanded query radius beyond the revised initial query radius if the number of neighbor objects is less than the target number;

locating neighbor objects within the revised expanded query radius;

determining if a number of neighbor objects within the revised expanded radius corresponds to the target number; and

continuing to expand the query radius and locate neighbor objects within the expanded radius until the number of neighbor objects equals or exceeds the target number.

12. (original) The method according to claim 11, wherein the minimum number of expansions is three.

13. (currently amended) The method according to claim 11, further comprising:

determining whether an increased number of tiles exceeds a memory capacity [[of]] in which the database is stored prior to increasing the number of tiles; and

if the increased number of tiles exceeds the memory capacity then redefining the tiles with an increased size.

14. (original) The method according to claim 7, further comprising:

calculating a distance of each located neighbor object from the query object if the number of neighbor objects in the initial query radius exceeds the target number to determine the target number of nearest neighbor objects.

15. (original) The method according to claim 7, wherein the tiles are rectangular.

16. (original) The method according to claim 7, wherein all of the tiles have the same shape and size.

17. (original) The method according to claim 7, wherein the database comprises locations in a geographical region, the query object represents a first location in the geographical region, and the neighbor objects represent other locations in the geographical region.

18. (cancelled)

19. (original) The method according to claim 7, further comprising:

assigning each tile a tile code and an associated geometry as the tiles are defined.

20. (original) The method according to claim 7, wherein the query radius is expanded a selected distance and the method further comprises determining a number of tiles necessary to expand the query radius the selected distance.

21. (original) The method according to claim 7, wherein the query radius is expanded a selected number of tiles.

22. (original) The method according to claim 7, wherein the initial query radius and the expanded query radius comprise the same number of tiles.

23. (original) The method according to claim 7, wherein the query object is grouped within a plurality of tiles.

24. (original) The method according to claim 23, wherein the query object is grouped completely or partially within the tiles.

25. (original) The method according to claim 7, wherein the database comprises a spatial or geographic database.

26. (original) The method according to claim 7, further comprising:

organizing the neighbor objects in order of distance.

27. (original) The method according to claim 7, further comprising:

calculating a distance of each located neighbor object from the query to determine the target number of nearest neighbor objects.

28. (original) The method according to claim 7, wherein the query object comprises a geometry that at least partially intersects at least one tile and the neighbor objects comprise geometries that at least partially intersect at least one tile.

29. (original) The method according to claim 28, wherein at least one tile that the neighbor objects intersect coincides with at least one tile that the query geometry intersects.

30. (currently amended) A computer program product for performing a process for locating neighbor objects of a query object in a database in a computer system, comprising:

a computer readable medium; and

computer program instructions, recorded on the computer readable medium, executable by a processor, for performing the steps of:

defining a query tile set comprising a plurality of tiles that define an initial query radius that the query object is grouped within [[;]] , wherein the query object is a non-point object;

locating neighbor objects among objects in the database within the initial query radius [[;]] , wherein the objects in the database are indexed using a linear quadtree index, and wherein the objects in the database include a non-point object;

determining a number of neighbor objects within the initial query radius;

expanding the query radius beyond the initial query radius by defining an expanded query tile set beyond the query tile set if the number of neighbor objects is less than a target number;

locating neighbor objects within the expanded query radius;

determining if a number of neighbor objects within the expanded radius corresponds to a target number; and

continuing to expand the query radius and locate neighbor objects within the expanded radius until the number of neighbor objects equals or exceeds the target number.

31. (currently amended) A system for performing a process for locating neighbor objects of a query object in a database, comprising:

a processor operable to execute computer program instructions; and

a memory operable to store computer program instructions executable by the processor, for performing the steps of:

defining a query tile set comprising a plurality of tiles that define an initial query radius that the query object is grouped within [[;]] , wherein the query object is a non-point object;

locating neighbor objects among objects in the database within the initial query radius [[;]] , wherein the objects in the database are indexed using a linear quadtree index, and wherein the objects in the database include a non-point object;

determining a number of neighbor objects within the initial query radius;

expanding the query radius beyond the initial query radius by defining an expanded query tile set beyond the query tile set if the number of neighbor objects is less than a target number;

locating neighbor objects within the expanded query radius;

determining if a number of neighbor objects within the expanded radius corresponds to a target number; and

continuing to expand the query radius and locate neighbor objects within the expanded radius until the number of neighbor objects equals or exceeds the target number.